#ANSIBLEFEST2019

Scaling Ansible for IoT

Brian Annis
Lead Site Reliability Engineer, Intersection

Red Hat | ANSIBLE

Share your automation story

1. How did you get started with Ansible?

2. How long have you been using it?

3. What's your favorite thing to do when you Ansible?


Intersection
Scaling Ansible for IoT
AnsibleFest 2019
September 2019

Agenda

1. Introduction
2. Challenges
3. Architecture
4. Deployment
5. Questions

1. Introduction

About Me

Brian Annis
Lead Site Reliability Engineer

About Intersection

Intersection connects the digital and physical worlds, enhancing people's journeys through their cities and offering brands the opportunity to drive more relevant and engaging advertising, rooted in real-world location and physical context.

Today IxN Engineering supports over 7000 IoT devices in dozens of cities around the globe

Link


The fastest Wi-Fi in NYC is free and it's right here.
Broadway


Master of None
Season 2
NETFLIX | MAY 12
USB POWER

What is Link?

1	Super fast, free gigabit Wi-Fi
2	Free phone calls
3	Interactive tablet for local search, wayfinding, and civic services
4	Access to emergency services and an emergency call button
5	Two 55” digital displays for advertising and messaging
6	Rapid USB charging
7	Free to residents, visitors, and workers, paid for by advertising

Transit


4:04
septa.org
MY COUCH NEEDED SOME SPACE
TOUGH MUDDER 5K · PHILLY
MAY 20, 2018
30% OFF · USE CODE: PHILLY5K
TOUGH MUDDER 5K
DIRECTIONS
DEPARTURES
STATUS


CANADA GOOSE
SPRING 2018

IxNTouch

Interactive Displays

Let customers explore real-time transit information (alerts, arrivals, transit agency messaging) alongside interactive features.

Key Features

- Interactive transit info
- Wayfinding, system maps, arrivals
- Service alerts
- Interactive advertising


On the Go Travel Station
ulton St
2:15 PM
5
Bronx
Directions
Itinerary Info
Points of Interest

MTA


2:59
NJTRANSIT
SEE Something,
SAY Something.
If It Doesn't Feel Right,
It Probably Isn't.
OBSERVE
If you see something suspicious or unusual, REPORT IT!
TEXT
Your message to NJTPD (65873) or call 1-888-TIPS-NJT
Text NJTPD or Call 1-888-TIPS-NJT
NJTRANSIT
DIRECTIONS
DEPARTURES
SYSTEM STATUS

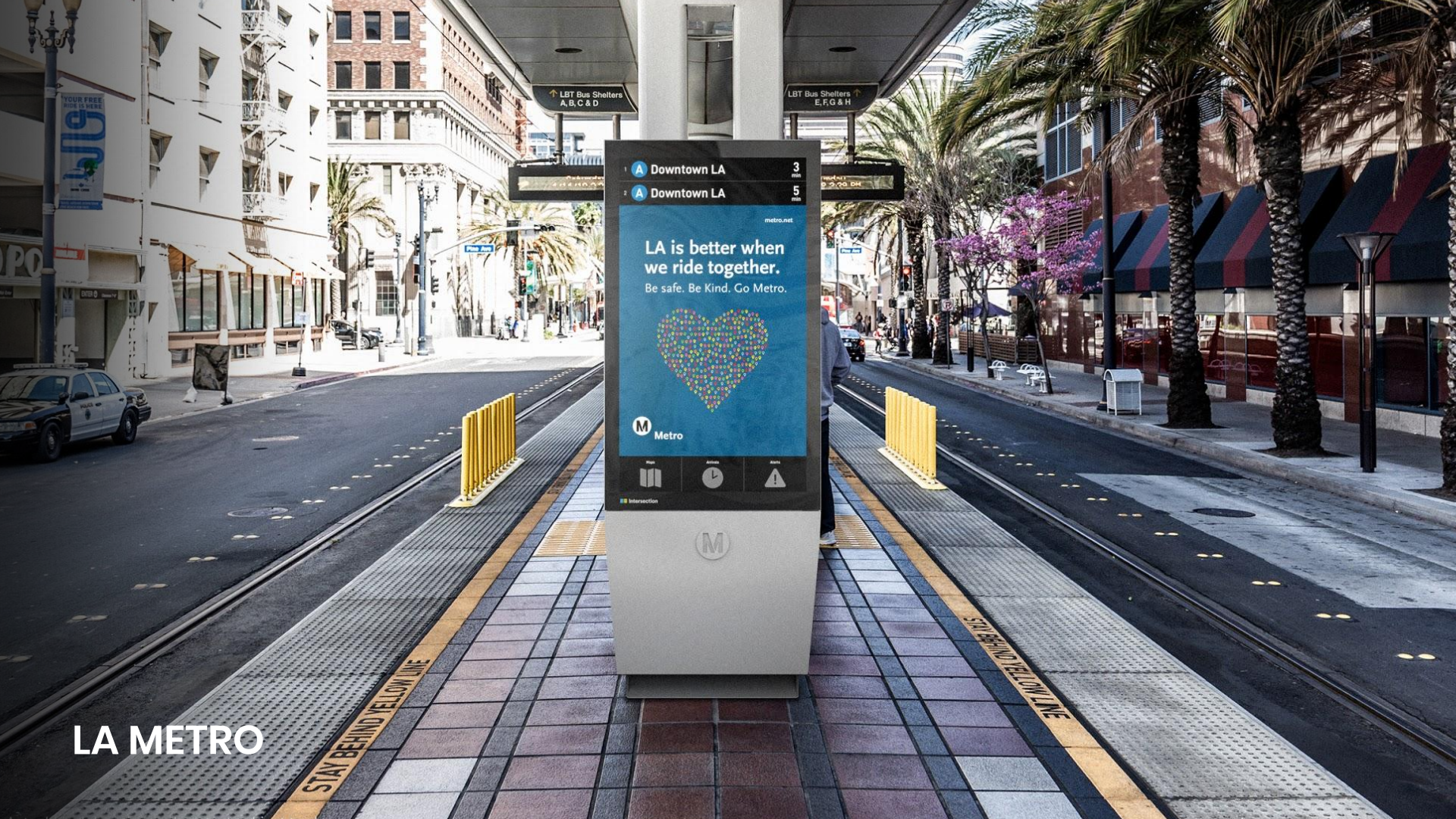
NJ TRANSIT


12:38
septa.org
Tap on a station to get directions and cost.
DIRECTIONS
DEPARTURES
SYSTEM STATUS

SEPTA


FAST-FIRE
BLAZE PIZZA
Howard
Due
Linden
Due
Brown Line
Purple Line
Elevator at Armitage Temporarily Out-of-Service
Elevator Status
The elevator to the Kimball- and Linden-bound platform at Armitage (Brown Line and Purple Line Express) is temporarily out-of-service.
7:52 am today
thru TBD
Belmont
cta Red, Purple and Brown Lines
GRUB

CTA


LBT Bus Shelters
A, B, C & D
LBT Bus Shelters
E, F, G & H
A Downtown LA 3 min
A Downtown LA 5 min
metro.net
LA is better when we ride together.
Be safe. Be Kind. Go Metro.
Metro
STAY BEHIND YELLOW LINE
STAY BEHIND YELLOW LINE

LA METRO

IxNSight

Dynamic Displays

Digital displays showing real-time transit information (alerts, arrivals, transit agency messaging) to customers alongside engaging advertising content.

Key Features

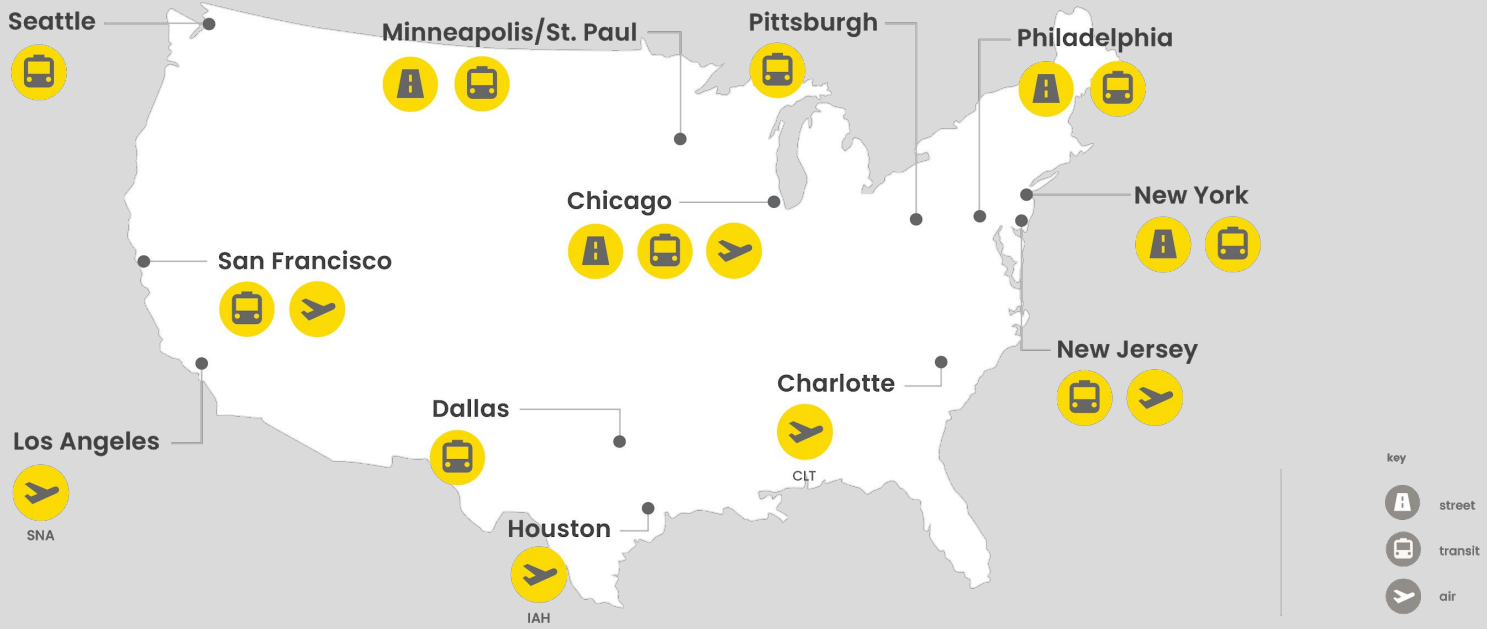
- Real-time arrival info
- Planned and real-time alerts
- Dynamic, targeted advertising
- TA visual standards


MAXIMO
Ashland 63rd 8 min
Cottage Grove 17 min
Harlem Lake 2 min
Harlem Lake 11 min
Ahead to north side of Cermak Rd and westbound buses

CTA

METRO
TRANSIT

Nationwide Digital Fleet



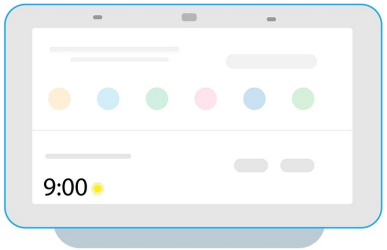
UK Digital Fleet

key

- street
- transit
- air

2. Challenges

Initial Assumptions



Low power SoC
Device is powered by a low power, embedded system on chip or single board computer

Live display
Device is required to display content, UI or other elements as part of its functionality

Heterogeneous Networks

Support multiple network topologies
Devices need to support a variety of connectivity options

No direct access
No inbound access via public or private networks

Hardware Variety

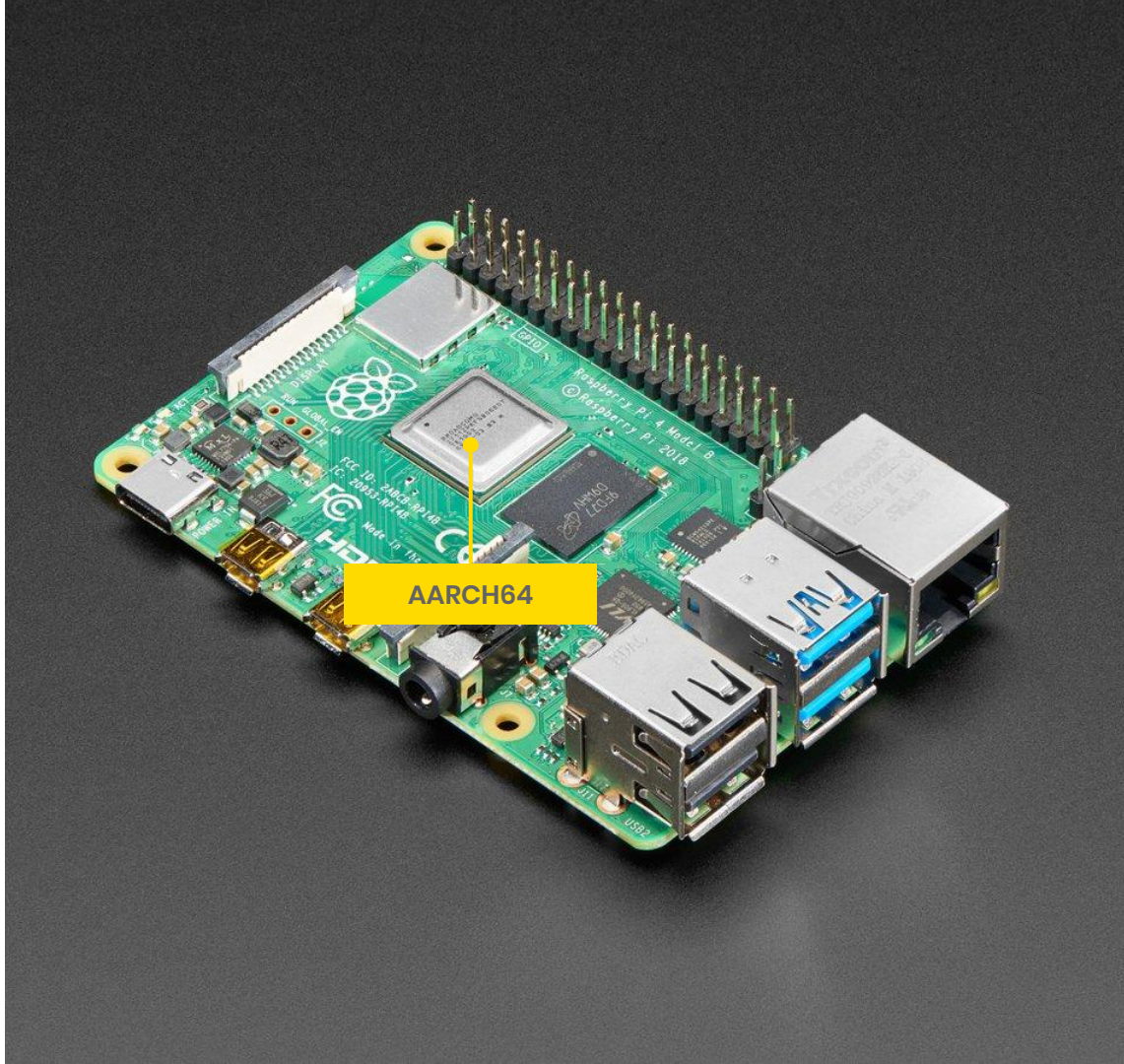
Multiple product generations
Street hardware has a service life of 10 years, while SBCs and accessories have relatively shorter availability periods

Variable hardware requirements
New deals may require takeover of TA's existing hardware fleet, adding to fragmentation

Single Board Computers

RTC availability
Embedded hardware may lack a real-time clock or battery backup

Alternate CPU architectures
SBCs may operate on x86 or ARM with varying degrees of mainline kernel / OS support



Security Requirements

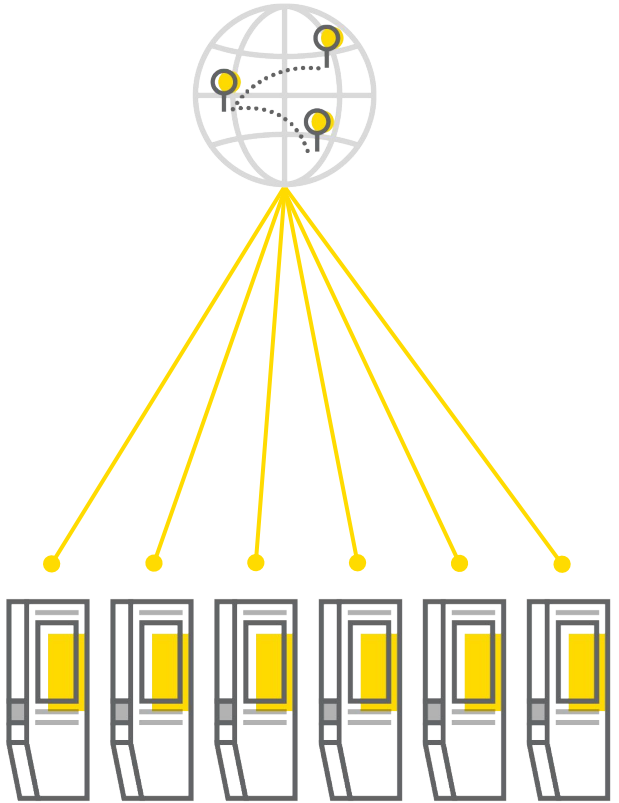
Shields up
Lock down physical, network, OS, and application stack across a diverse operating environment

Plan for a hostile world
Assume that the environment is public and that all data must be encrypted

Traffic Management

Manage fleet traffic
Prevent DDoS of services by tranching and enabling jitter across the footprint

Eventual consistency
Real time coordination across multiple network and geographic partitions is not required



Screens as a Service

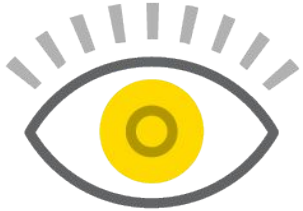
Operate in physical space
Understand that outages may be out of your control: physical damage, water ingress, network congestion

Adapt monitoring for real displays
Operate thousands of backlights, LCD panels, display controllers and X sessions in the real world

Beyond these challenges, Intersection also faced specific legacy constraints

Time Sync

Unreliable OS time on some devices



Visibility

Limited metrics on deployment status, fleet uptime, or device health

Release Testing

Lack of automated, end to end testing with hardware in realistic environments

Challenges
Legacy Constraints

SBC Hardware

- No RTC or CMOS battery
  - Embedded board was not populated with RTC chip
  - Can't rely on TLS for CM
- Existing CM solution required TLS

Visibility

- No global metrics platform
  - Release visibility limited to log queries
  - Hard to gauge success of release
- Trailing alerting
  - Alerting based on queries that may take minutes to run

Challenges
Recap

- Thousands of low power computers
- Dozens of discrete networks
- Operating globally in public space
- Equipped with displays
- Occasionally January 1st, 1970

3. Architecture

Architecture
Requirements

1

Should be fully managed in a GitOps compliant way

2

Ensures transit encryption, even with incorrect device time

3

Can scale to thousands of nodes across multiple products and geographic partitions

4

Undergoes multiple automated SDLC test stages, including unit and integration tests

It’s time for something a little bit different.

While Ansible is traditionally used in push mode, the SRE team decided to deploy Ansible in pull mode to meet unique architecture requirements.

Architecture
Ansible Pull

ansible-pull is used to clone a repository containing ansible assets and execute them.

This wrapper enables ansible to operate in pull mode, which has near-limitless scaling potential.

Architecture
Ansible Pull

Checkout Targets

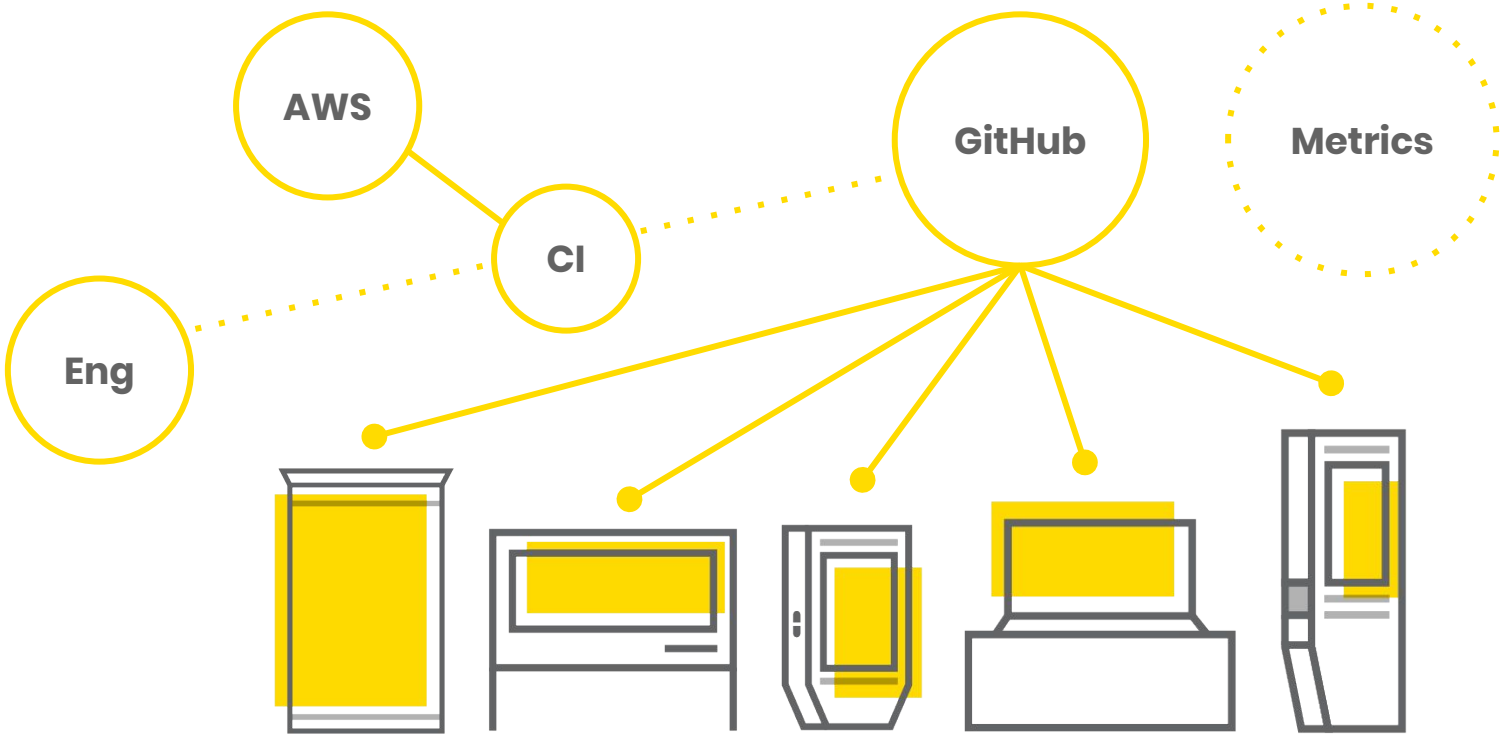
- Branch
- Tag
- Commit ID

Release Management

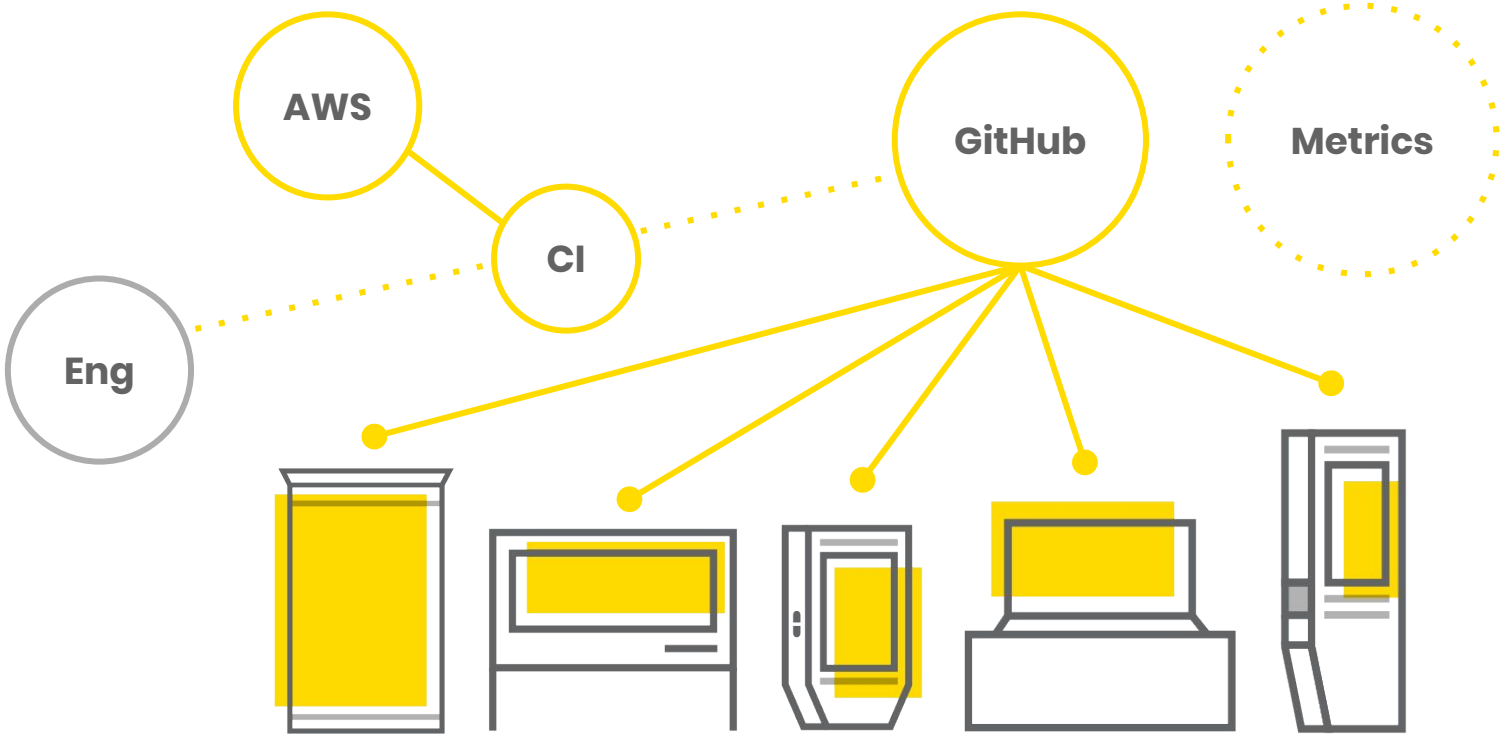
- Jitter
- Clone depth
- Directory cleanup

Architecture

Architecture



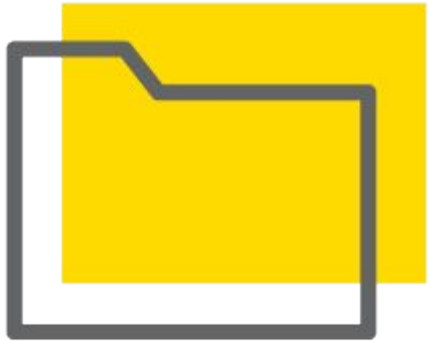
Architecture



Architecture
Inventory

One inventory per env_zone
Target localhost and use alternate directory based inventory structure

Use metadata service to steer
Metadata service writes local facts which are consumed by update service



Architecture
Metadata

Gather device metadata
Poll external metadata sources and write native facts to be consumed by ansible

Enable failsafe execution
Decoupled metadata service allows ansible to utilize fact cache even when there is no service availability

```
[global]
branch=master
zone=ne
env=dev
role=ad
product=transit
customer=ta
site=zz-123456
```

Architecture
Transport Encryption

Secure operations without TLS
Use RSA keys to access GitHub and clone repository

One way transactions
Read-only access prevents reverse supply chain attacks

Architecture
Run Scheduling

Native scheduling
Leverage systemd services and timers to ensure eventual consistency and error handling

Traffic management
Employ jitter and timers to ensure services are not DDoS'd by fleet

Architecture

```
1 [Unit]
2 Description=Ansible Pull
3 OnFailure=ap-reset.service
4 Wants=ap-updater.service
5 Before=ap-updater.service
6
7 [Service]
8 ENVIRONMENT=JITTER=60s
9 ExecStart=ansible-pull -i {{ inventory }} -C {{ branch }} -U git@github.com:org/repo.git --sleep
  $JITTER playbook.yml
10 ...
11
12 [Install]
13 WantedBy=a-target.target
```

Architecture
Recovery

Activate self healing
Engage reset service to check device health and remediate

Ensure run capability
Purge repository directory on run error to prevent git problems from halting agent

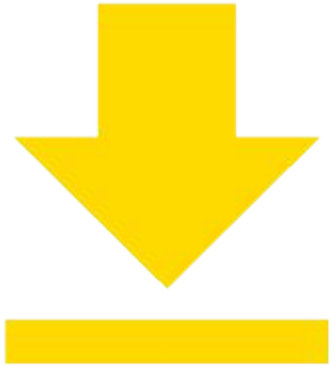
Architecture

```
1 #!/bin/bash
2 echo "Clearing local Ansible repo..."
3 rm -rf /opt/.ansible/pull
4
5 echo "Checking for and repairing venv corruption..."
6 if [ $(/opt/venv/bin/pip2 | wc -l) -eq 0 ]; then rm -rf /opt/venv; fi
7 ...
8
9
```

Architecture
Upgrades

Provide self upgrade capability
Manage ansible upgrades with mutually exclusive systemd service

Use existing templating engine
Write ansible service file and update according to local facts with Jinja2

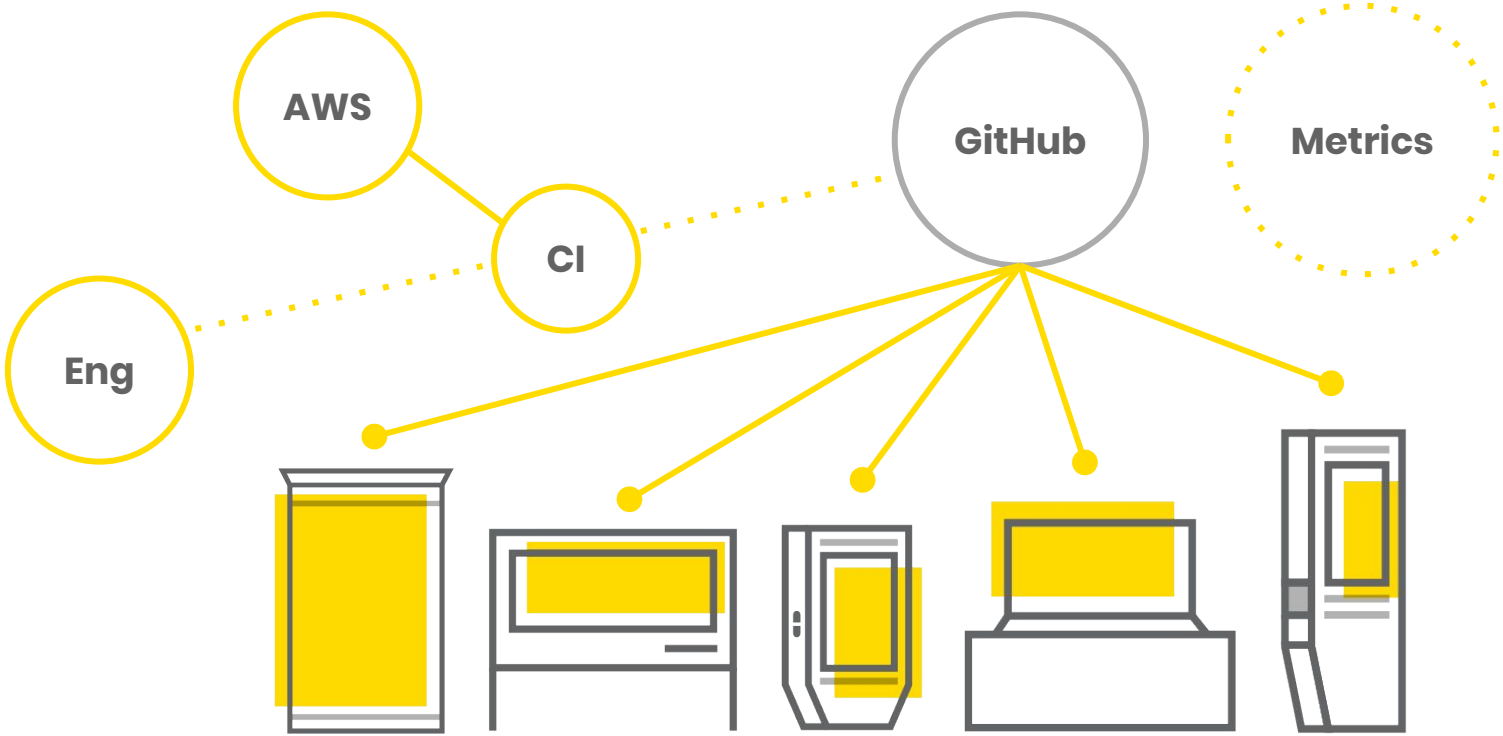


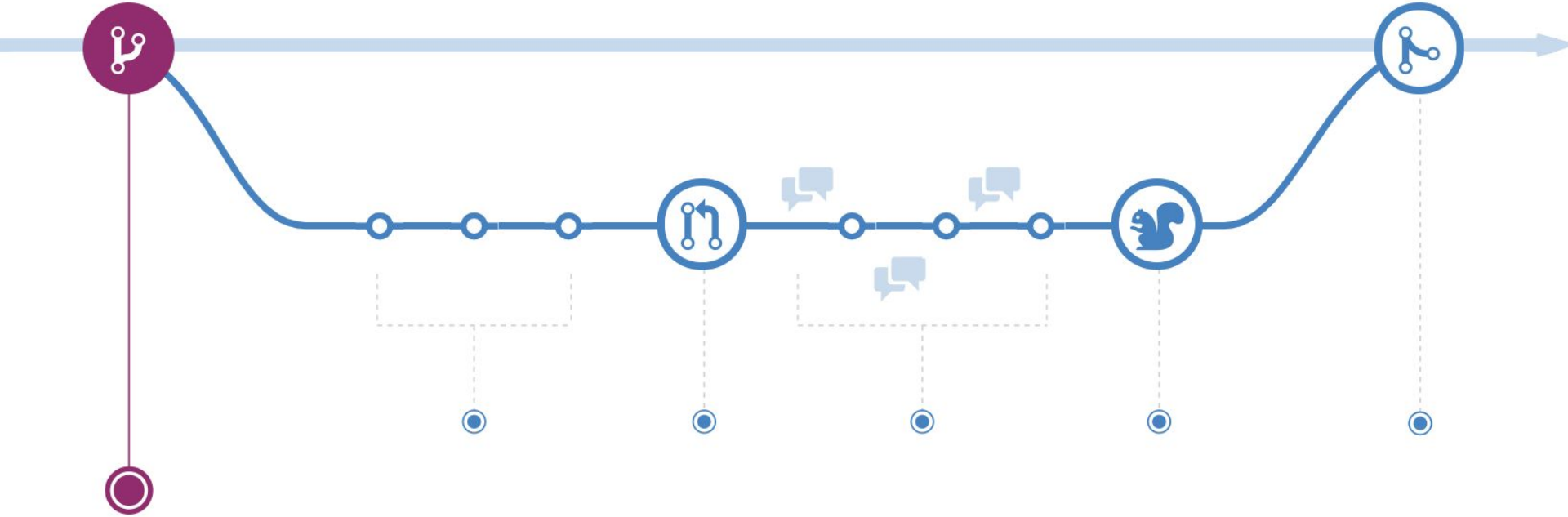
Architecture
Why Ansible Pull

Pull Architecture
Securely manage distributed devices without inbound access or narrow connectivity requirements

Scaling
Ansible Pull scales to thousands of nodes with highly predictable traffic patterns

Architecture





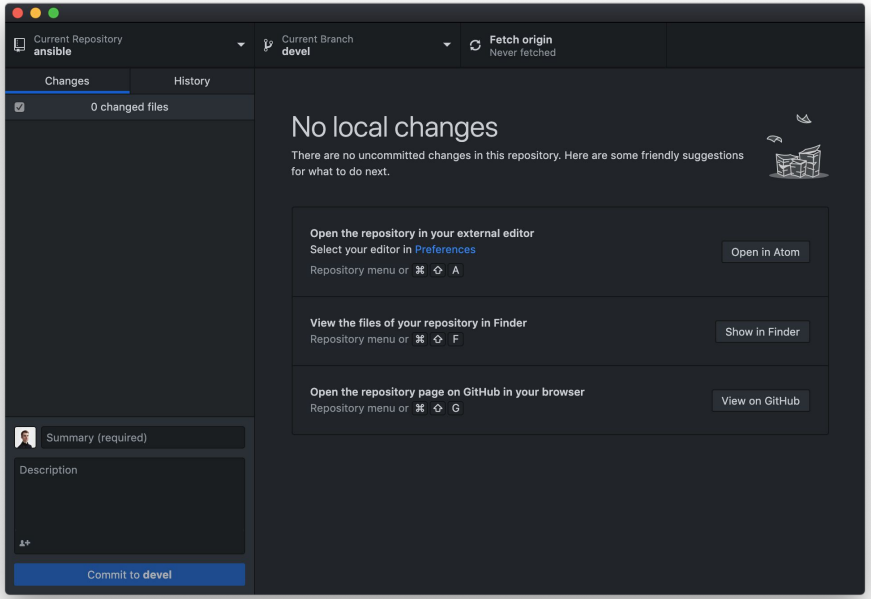
What is GitHub Flow?

GitHub flow is a lightweight, branch-based workflow that supports teams and projects where deployments are made regularly.

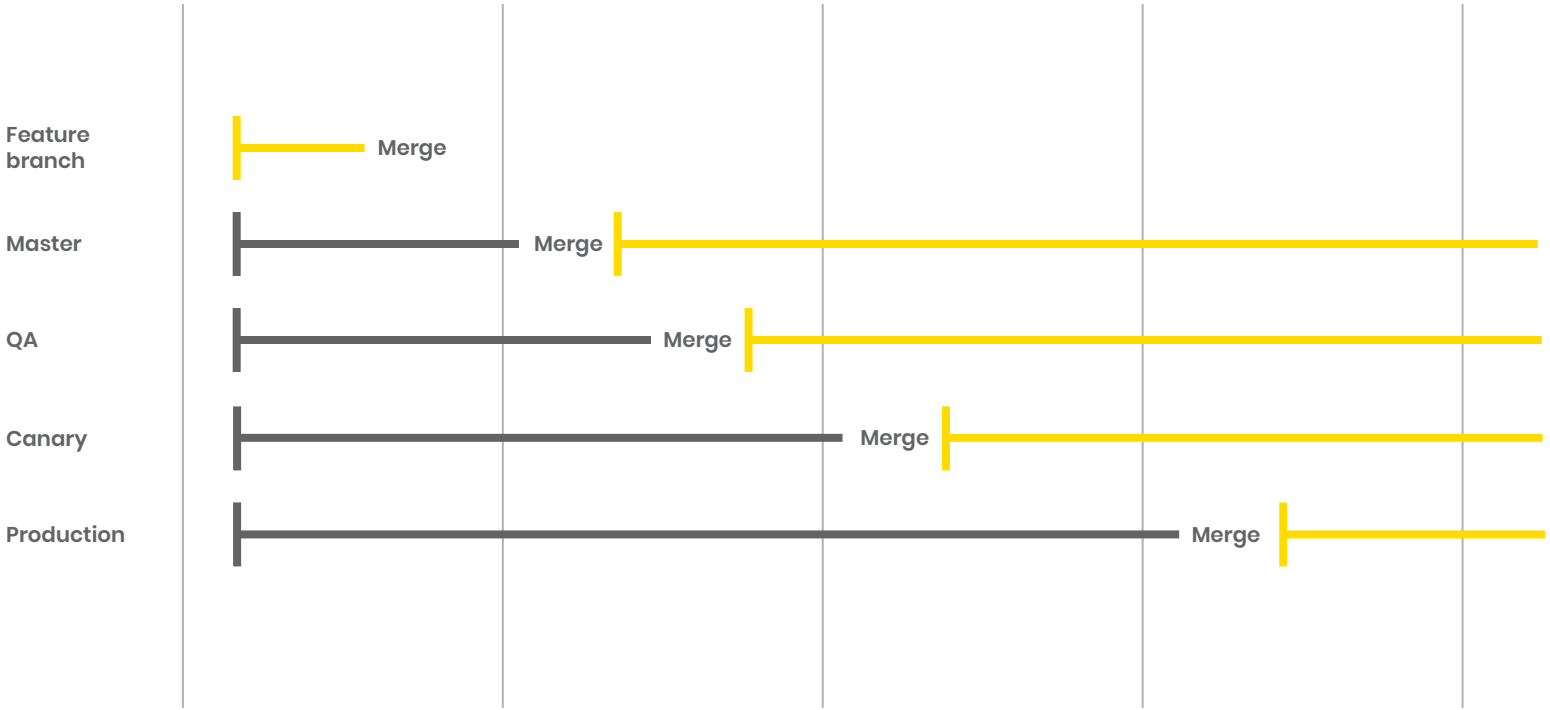
Architecture
Git Workflow

Feature branches
Create a branch for each atomic feature to be written and tested

Local development environment
Branch based workflow means that it is trivial to create a “development environment”



Branching Strategy



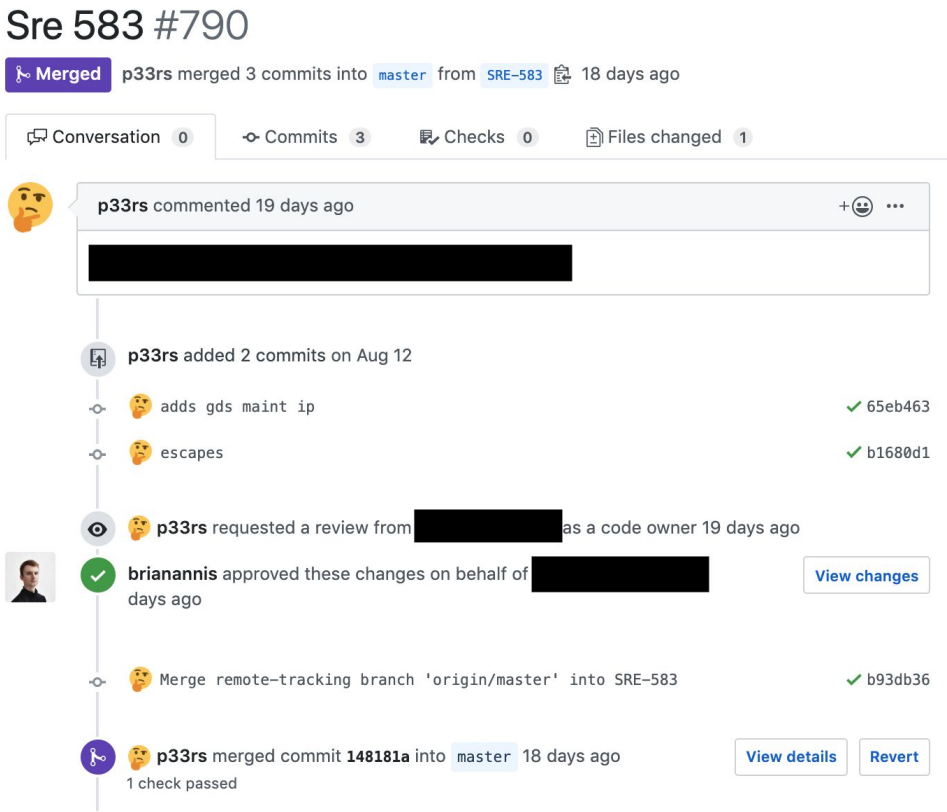
Architecture
Git Workflow

Pull requests required

PRs can only be merged to master after passing tests and being approved by the SRE team

CI manages releases

The CI platform is the only entity that can merge to upstream branches

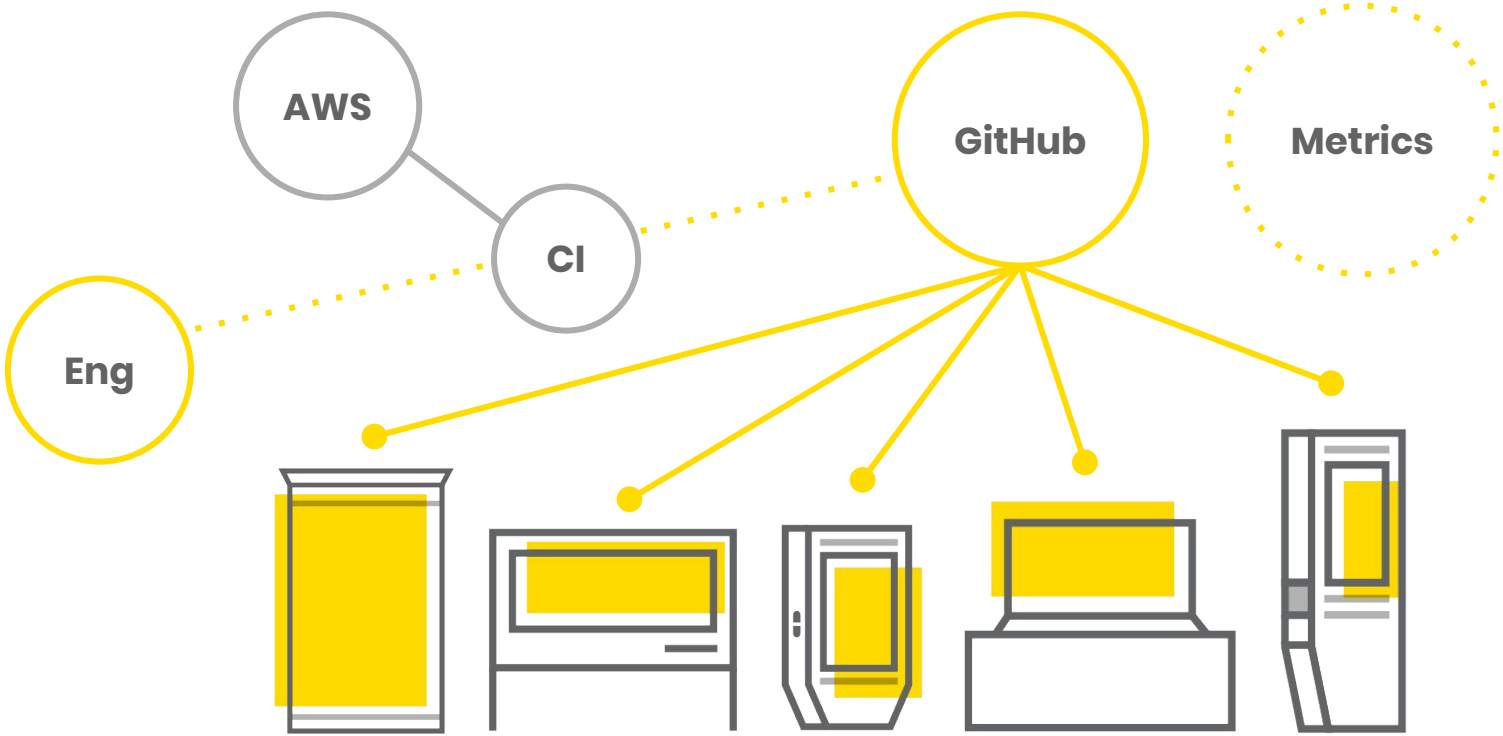


Architecture
Why Ansible Pull

GitOps
Manage infrastructure as code using GitHub Flow

Environment visibility
Understand the current state of the environment by inspecting the corresponding branch

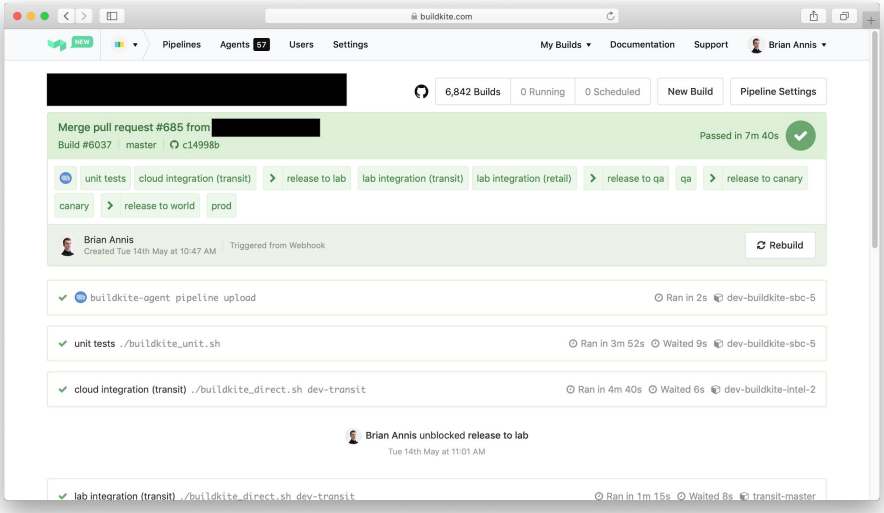
Architecture



Architecture
Buildkite

Buildkite is responsible for running test and release jobs though a dedicated pipeline.

- Each commit initiates unit tests and integration tests
- Upon merging to master, CI controls the release through to production

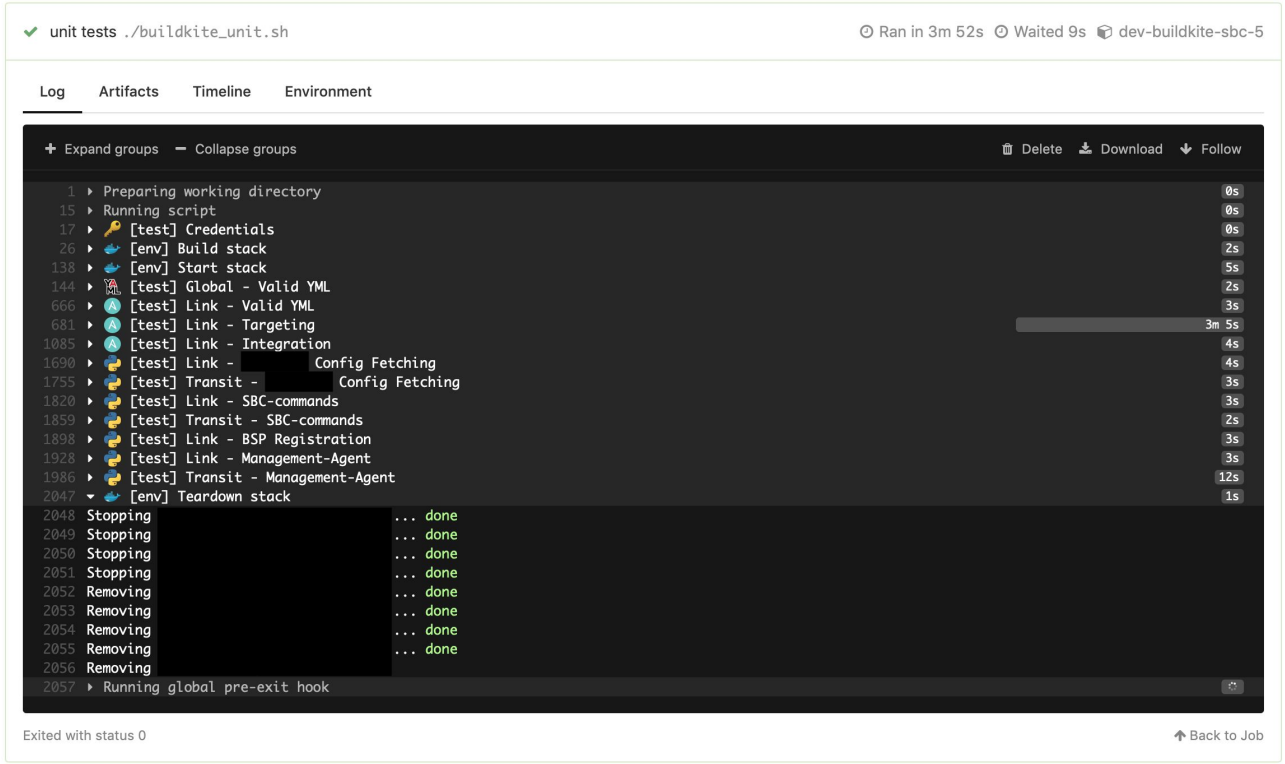


Architecture
Test Suite

Linting + Validation
Test each commit for valid YAML and defined variables

Unit Tests
Run unit tests for Python code that augments platform functionality

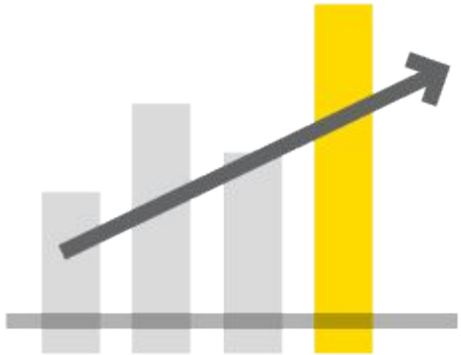
Architecture



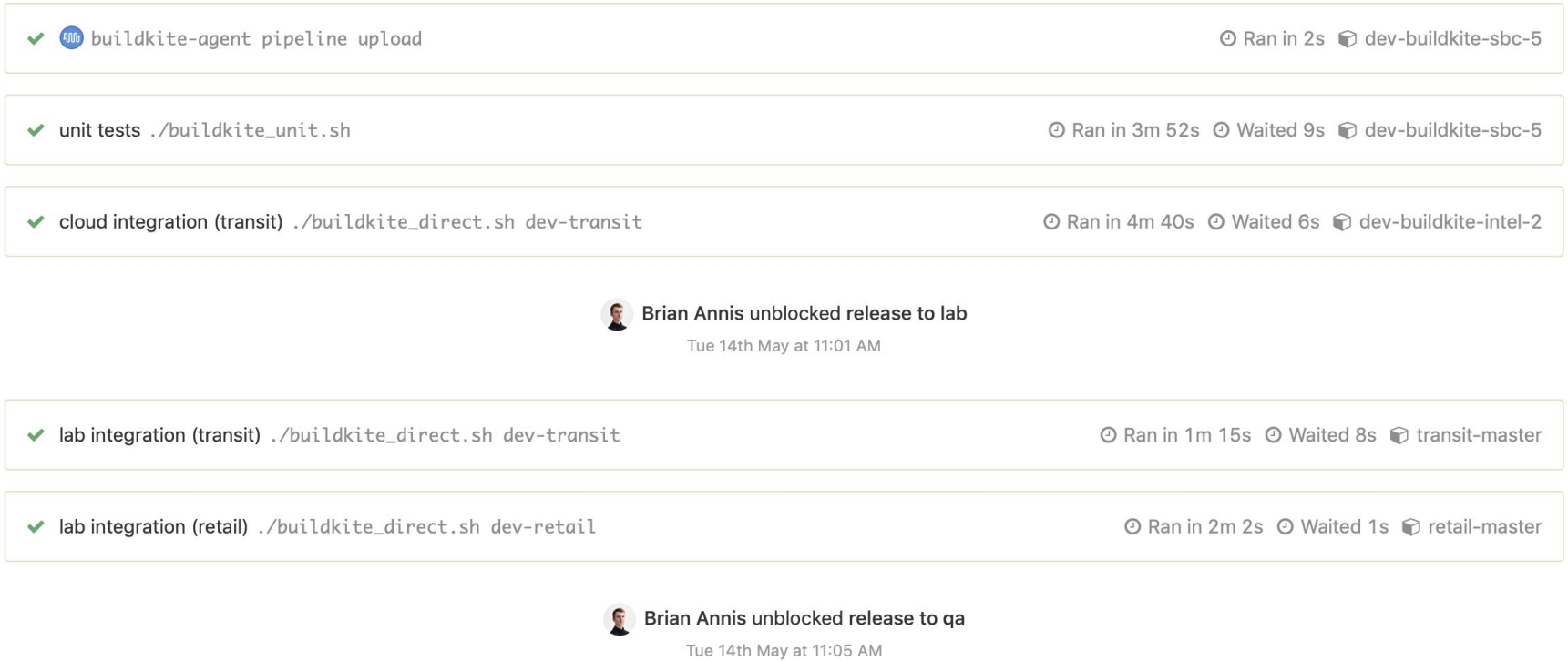
Architecture
Test Suite

Integration Tests
Run ansible playbook end to end on disposable EC2 instance

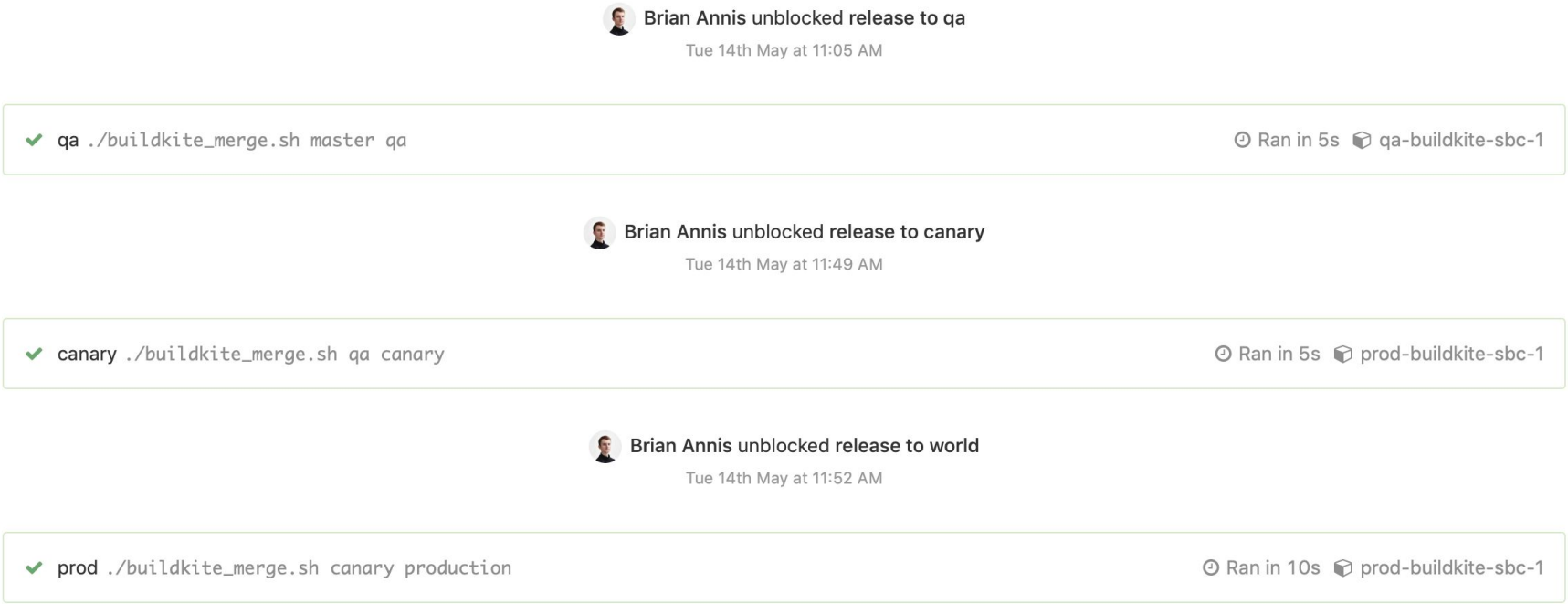
Lab Test
Run ansible playbook on hardware devices that are used in the field



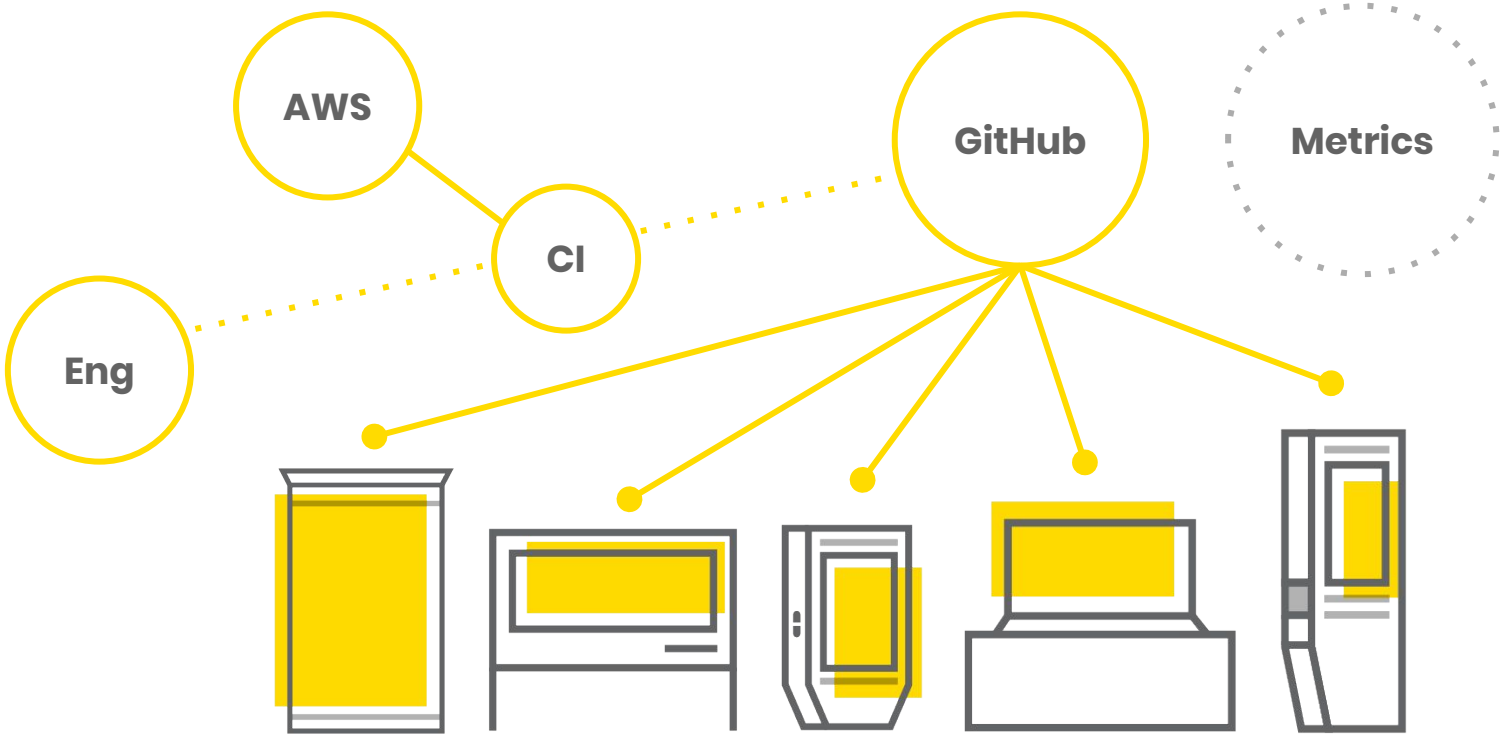
Architecture



Architecture



Architecture

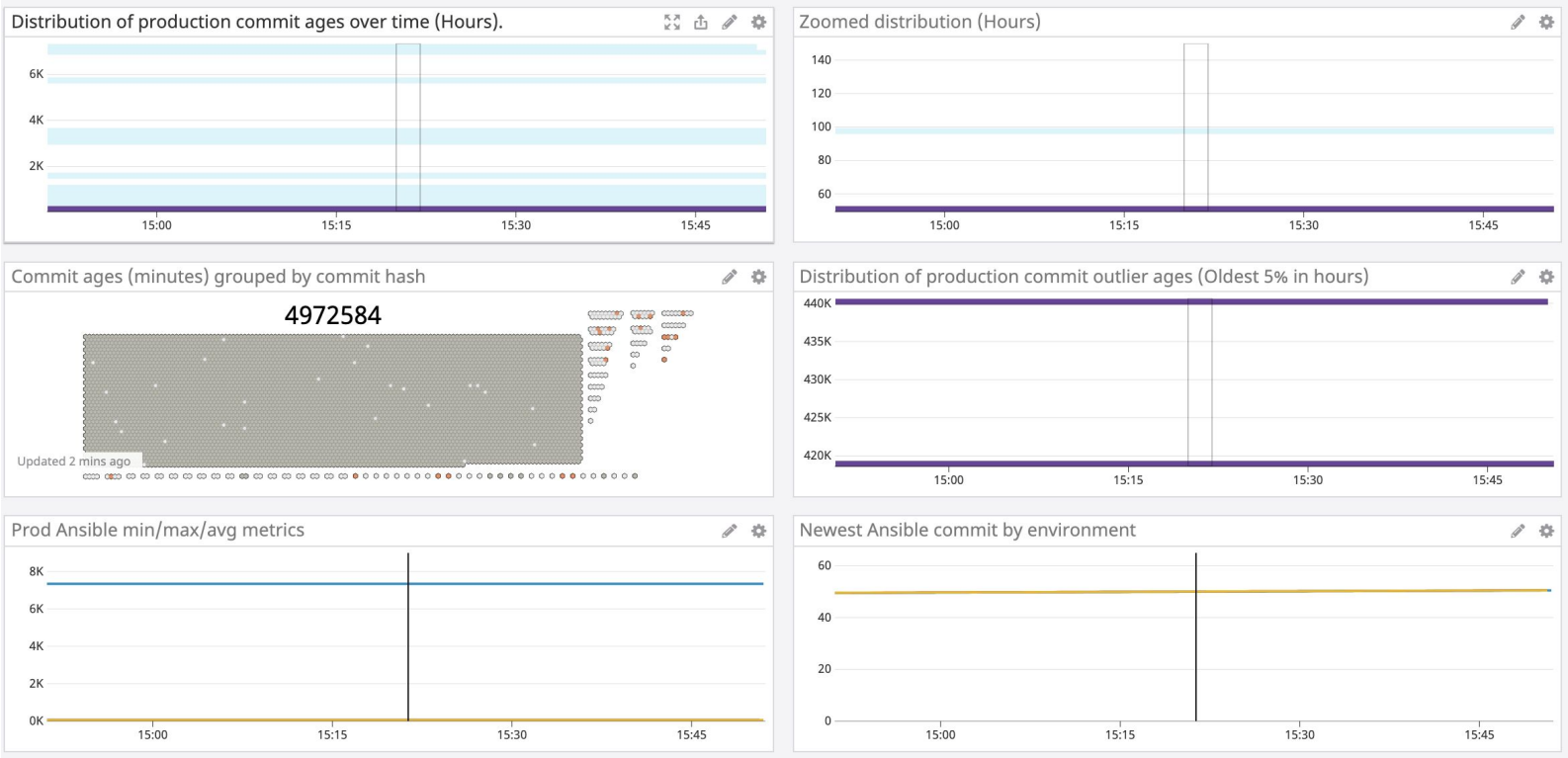


Architecture
Datadog

Release events
Buildkite release triggers event notification to Datadog

Release monitoring
Monitor commit age, number of devices on commit, and health in real time

Architecture



Architecture
Visibility

Native reporting
Leverage the ansible Datadog module to send the current state of the run and status code

Surfacing outliers
Find and triage devices that are failing the run, stuck on an old commit, or offline.

Architecture
Alerting

Fleetwide status
Create monitors to report fleetwide ansible completion status and alert when when degraded

Component checks
Write Datadog “checks” that report individual component status to calculate product health

Architecture
Recap

1

A GitOps compliant architecture

2

Utilizes RSA keys to ensure transport security without TLS

3

Can scale to thousands of nodes distributed across the globe

4

Each release is tested & validated by a CI pipeline

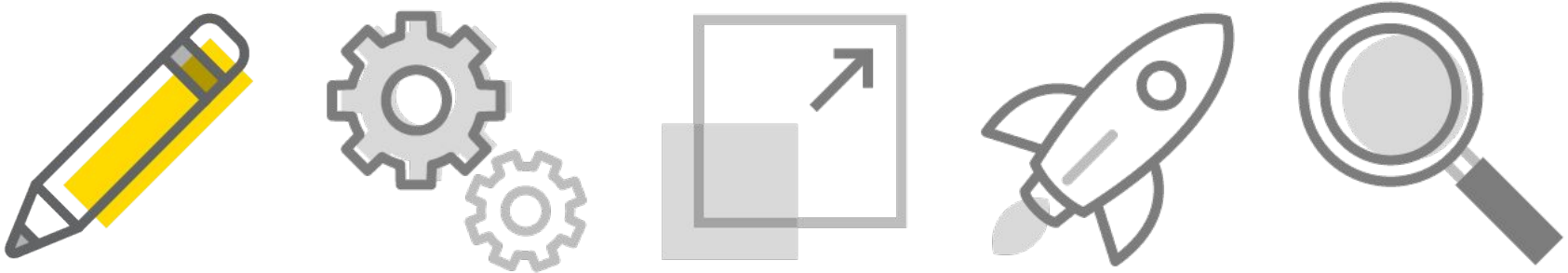
4. Deployment

With the IoT, we're headed to a world where things aren't liable to break catastrophically – or at least we'll have a hell of a heads' up.

Scott Wiess, General Partner
Andreessen Horowitz

Deployment
Release Train

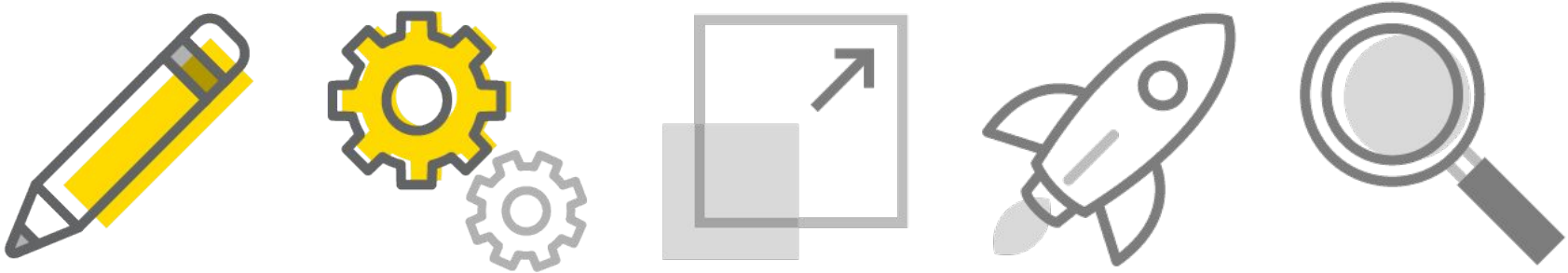
A feature branch is created and new functionality is added to it



Commit **Test** **Merge** **Release** **Monitor**

Deployment
Release Train

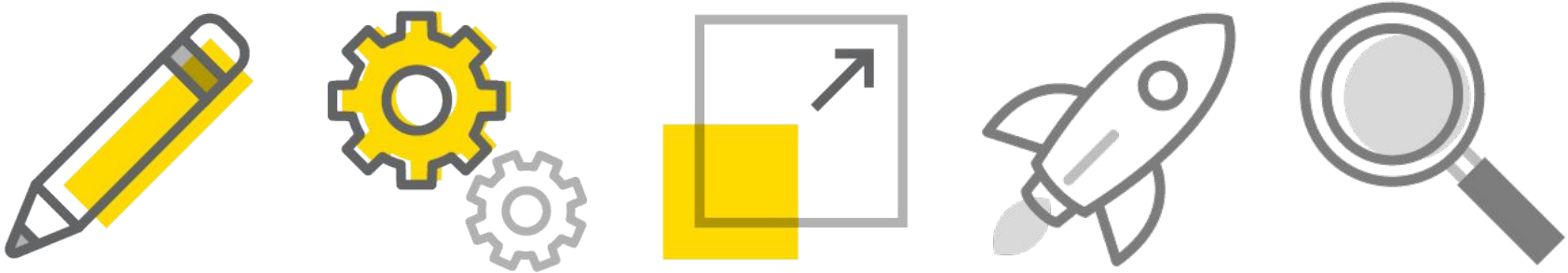
Commits to this branch kick off unit and integration tests



Commit **Test** **Merge** **Release** **Monitor**

Deployment
Release Train

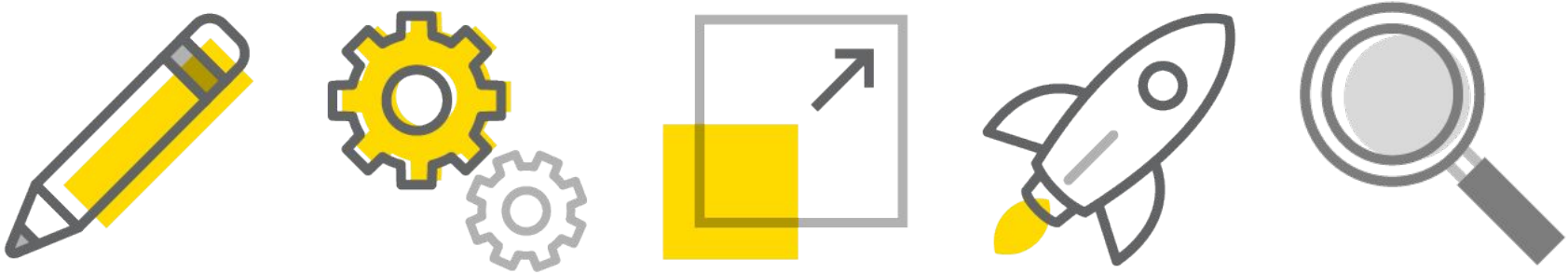
Once merged, the release blocks the pipeline until the change set is live or superceded



Commit Test Merge Release Monitor

Deployment
Release Train

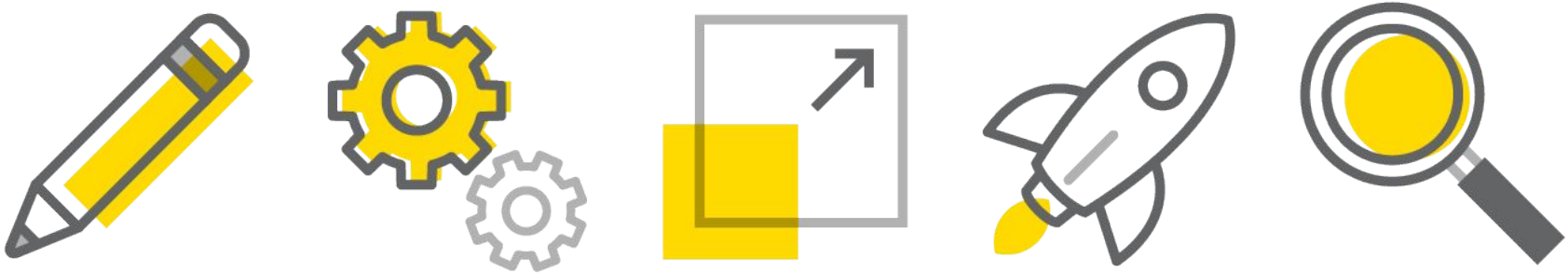
As the change set is validated in each environment it is released to the next stage



Commit Test Merge Release Monitor

Deployment
Release Train

Finally, the release metrics are monitored and tracked by Datadog



Commit Test Merge Release Monitor

Deployment
Velocity

50
Releases / month

3885
Commits

22
Contributors

Source: GitHub

We're Hiring

Drive innovation and technology for the future of cities: **intersection.com/join-our-team**

5. Questions

Intersection

Thank You

Intersection
10 Hudson Yards 26th Floor
New York NY 10001
intersection.com

Brian Annis
Lead Site Reliability Engineer
brian.annis@intersection.com

#ANSIBLEFEST2019

THANK YOU


youtube.com/AnsibleAutomation


facebook.com/ansibleautomation


linkedin.com/company/Red-Hat


twitter.com/ansible